AMENDMENTS TO THE CLAIMS

The following listing of claims will replace all prior versions and listings of claims in the application.

LISTING OF CLAIMS

1. (Currently Amended) A line cache control system that controls data flow between a line cache, a first central processing unit (CPU) and first and second memory devices, comprising:

a first line cache interface that is associated with the first CPU, that receives a first program read request from the first CPU and that generates a first address from said first program read request;

a first memory interface that communicates with the first memory device;

a second memory interface that communicates with the second memory device;

a line cache that receives a second address that is based on the first address and includes a memory select portion; and

a switch that receives the second address and that selectively connects said line cache to one of said first and second memory interfaces based on the memory select portion,

wherein when said line cache receives said ~~first~~ second address, said line cache compares said ~~first~~ second address to stored addresses in said line cache, returns data associated with said ~~first~~ second address if a match occurs, and retrieves data from one of the first and second memory devices if a miss occurs, and

wherein said switch includes a plurality of selectors that each receive the second address and each select between first and second sets of signals relating to the first and second memory devices, respectively, based on the second address.

2. (Previously Presented) The line cache control system of claim 1 wherein said first memory device is random access memory (RAM).

3. (Previously Presented) The line cache control system of claim 2 wherein said RAM is one of dynamic RAM (DRAM), synchronous DRAM (SDRAM), and double data rate SDRAM (DDRAM).

4. (Original) The line cache control system of claim 1 wherein said second memory device is flash memory.

5. (Original) The line cache control system of claim 1 wherein said first CPU is an advanced risc machine (ARM) processor.

6. (Currently Amended) The line cache control system of claim 1 further comprising:

a second CPU;

a second line cache interface that is associated with said second CPU, that receives a second program read request from said second CPU and that generates a ~~second~~ third address from said second program read request; and

a line cache arbitration device that communicates with said first and second line cache interfaces and said line cache and that resolves line cache access conflicts between the first CPU and said second CPU.

7. (Original) The line cache control system of claim 6 further comprising:

a first direct interface that is associated with the first CPU,

wherein said first memory interface includes a second direct interface that communicates with said first direct interface and wherein said first and second direct interfaces allow the first CPU to at least one of read and write data directly to the first memory device.

8. (Original) The line cache control system of claim 7 further comprising:

a third direct interface that is associated with the second CPU,

wherein said second memory interface includes a fourth direct interface that communicates with said third direct interface and wherein said third and fourth direct interfaces allow said second CPU to at least one of read and write data directly to the second memory device.

9. (Original) The line cache control system of claim 8 further comprising a direct read/write arbitration device that resolves direct memory access conflicts between said first and third direct interfaces.

10. (Original) The line cache control system of claim 6 wherein the first CPU is a host processor for a hard disk drive and said second CPU is a servo processor for said hard disk drive.

11. (Original) The line cache control system of claim 1 wherein said line cache includes:

line cache memory that stores data;

a content addressable memory (CAM) that stores addresses associated with said data stored in said line cache memory; and

a line cache module that includes a line cache state machine that determines when one of a hit and a miss occurs and that manages retrieval of data from the first and second memory devices when said miss occurs.

12. (Currently Amended) A line cache control system that controls data flow between a line cache, a first central processing unit (CPU), a second CPU, and first and second memory devices, comprising:

a first line cache interface that is associated with the first CPU, that receives a first program read request from the first CPU and that generates a first address from said first program read request;

a first memory interface that communicates with the first memory device;

a second memory interface that communicates with the second memory device;

a line cache; and

a switch that selectively connects said line cache to one of said first and second memory interfaces, wherein when said line cache receives said first address, said line cache compares said first address to stored addresses in said line cache, returns data associated with said first address if a match occurs, and retrieves data from one of the first and second memory devices if a miss occurs,

wherein said line cache includes:

line cache memory that stores data;

a content addressable memory (CAM) that stores addresses associated with said data stored in said line cache memory; and

a line cache module that includes a line cache state machine that determines when one of a hit and a miss occurs and that manages retrieval of data from the first and second memory devices when said miss occurs; and

wherein said line cache memory includes multiple pages and wherein said line cache module allows one page to be accessed by one of the first CPU and said second CPU while the other of the first CPU and said second CPU is waiting for data retrieval in another page,

wherein said switch includes a plurality of selectors that each receive a second address associated with the first address and each select between first and second sets of signals relating to the first and second memory devices, respectively, based on the second address.

13. (Original) The line cache control system of claim 11 further comprising a least used page device that identifies a least used page in said line cache.

14. (Original) The line cache control system of claim 13 wherein said least used page device replaces said least used page with data retrieved from one of the first and second memory devices when a miss occurs.

15. (Previously Presented) The line cache control system of claim 11 wherein state transitions of said line cache state machine are based, in part, on at least one internal state of the first CPU.

16. (Currently Amended) A line cache control system that controls data flow between a line cache, a first central processing unit (CPU) and first and second memory devices, comprising:

a first line cache interface that is associated with the first CPU, that receives a first program read request from the first CPU and that generates a first address from said first program read request;

a first memory interface that communicates with the first memory device;

a second memory interface that communicates with the second memory device;

a line cache;

a switch that selectively connects said line cache to one of said first and second memory interfaces, wherein when said line cache receives said first address, said line cache compares said first address to stored addresses in said line cache, returns data associated with said first address if a match occurs, and retrieves data from one of the first and second memory devices if a miss occurs;

a least used page device that identifies a least used page in said line cache, that replaces said least used page with data retrieved from one of the first and second memory devices when a miss occurs, and that identifies a second least used page and wherein said line cache state module checks internal states of the first CPU,

wherein said line cache includes:

line cache memory that stores data;

a content addressable memory (CAM) that stores addresses associated with said data stored in said line cache memory; and

a line cache module that includes a line cache state machine that determines when one of a hit and a miss occurs and that manages retrieval of data from the first and second memory devices when said miss occurs,

wherein said switch includes a plurality of selectors that each receive a second address based on the first address and each select between first and second sets of signals relating to the first and second memory devices, respectively, based on the second address.

17. (Previously Presented) The line cache control system of claim 16 wherein said least used page is replaced when a miss occurs and internal states of the first CPU do not indicate a likelihood that said least used page will be needed within a predetermined period.

18. (Previously Presented) The line cache control system of claim 17 wherein said second least used page is replaced when a miss occurs and internal states of the first CPU indicate a likelihood that said least used page will be needed within a predetermined period.

19. (Currently Amended) A line cache control system that controls data flow between a line cache, a first central processing unit (CPU) and first and second memory devices, comprising:

a first line cache interface that is associated with the first CPU, that receives a first program read request from the first CPU and that generates a first address from said first program read request;

a first memory interface that communicates with the first memory device;

a second memory interface that communicates with the second memory device;

a line cache; and

a switch that selectively connects said line cache to one of said first and second memory interfaces, wherein when said line cache receives said first address, said line cache compares said first address to stored addresses in said line cache, returns data associated with said first address if a match occurs, and retrieves data from one of the first and second memory devices if a miss occurs,

wherein said first CPU executes an application and wherein said line cache has a line width and number of pages that are based on said application.

wherein said switch includes a plurality of selectors that each receive a second address based on the first address and each select between first and second sets of signals relating to the first and second memory devices, respectively, based on the second address.

20. (Previously Presented) The line cache control system of claim 1 wherein said line cache includes 4 pages of 8 x 32 bits.

21. (Currently Amended) A memory control system for a line cache and first and second memory devices that are accessed by a first central processing unit (CPU) and a second CPU, comprising:

a first line cache interface that is associated with the first CPU, that receives a first program read request from the first CPU and that generates a first address from said first program read request;

a second line cache interface that is associated with the second CPU, that receives a second program read request from the second CPU and that generates a second address from said second program read request;

a line cache arbitration device that communicates with said first and second line cache interfaces and said line cache that resolves line cache access conflicts between the first CPU and the second CPU, and that generates a translated address based on one of the first address and the second address that includes a memory select portion; and

a switch that receives the translated address and that selectively connects said line cache to one of said first and second memory devices based on the memory select portion,

wherein said switch includes a plurality of selectors that each receive the translated address and each select between first and second sets of signals relating to the first and second memory devices, respectively, based on the translated address.

22. (Previously Presented) The memory control system of Claim 21 wherein said line cache arbitration device selects one of said first and second addresses and further comprising:

a first memory interface that communicates with the first memory device; and

a second memory interface that communicates with the second memory device, wherein when said line cache receives said selected one of said first and second addresses from said line cache arbitration device, said line cache compares said selected one of said first and second addresses to stored addresses in said line cache, returns data associated with said selected one of said first and second addresses if a match occurs, and retrieves data from one of the first and second memory devices if a miss occurs.

23. (Previously Presented) The memory control system of claim 21 wherein the first memory device is random access memory (RAM) and the second memory device is flash memory.

24. (Previously Presented) The memory control system of claim 21 wherein the first CPU is an embedded processor.

25. (Previously Presented) The memory control system of claim 21 further comprising:

a first direct interface that is associated with the first CPU, wherein said first memory interface includes a second direct interface that communicates with said first direct interface and wherein said first and second direct interfaces allow the first CPU to at least one of read and write data directly to the first memory device; and

a third direct interface that is associated with the second CPU, wherein said second memory interface includes a fourth direct interface that communicates with said third direct interface and wherein said third and fourth direct interfaces allow the second CPU to at least one of read and write data directly to the second memory device.

26. (Previously Presented) The memory control system of claim 25 further comprising a direct read/write arbitration device that resolves direct memory access conflicts between said first and third direct interfaces.

27. (Previously Presented) The memory control system of claim 21 wherein the first CPU is a host processor for a hard disk drive and the second CPU is a servo processor for said hard disk drive.

28. (Previously Presented) The memory control system of claim 21 wherein said line cache includes:

line cache memory that stores data;

a content addressable memory (CAM) that stores addresses associated with said data stored in said line cache memory; and

a line cache module that determines when one of a hit and a miss occurs and that manages retrieval of data from the first and second memory devices when said miss occurs.

29-43 (Cancelled).

44. (Currently Amended) A method for operating a line cache, comprising:

receiving a first program read request from a first CPU at a first line cache interface;

generating a first address from said first program read request;

generating a translated address that includes a memory select portion based on said first address;

selectively connecting said line cache to one of first and second memory interfaces for first and second memory devices, respectively, based on said memory select portion, wherein said selectively connecting includes selectively connecting said line cache to said one of said first and second memory interfaces with a plurality of selectors that each receive the translated address and each select between first and

second sets of signals relating to the first and second memory devices, respectively, based on the translated address;

comparing said first address to stored addresses in said line cache when said line cache receives said first address;

returning data associated with said first address if a match occurs; and

retrieving data from one of said first and second memory devices if a miss occurs.

45. (Previously Ppresented) The method of claim 44 wherein said first memory device is random access memory (RAM).

46. (Previously Presented) The method of claim 45 wherein said RAM is one of dynamic RAM (DRAM), synchronous DRAM (SDRAM), and double data rate SDRAM (DDRAM).

47. (Previously Presented) The method of claim 44 wherein said second memory device is flash memory.

48. (Previously Presented) The method of claim 44 wherein said first CPU is an embedded processor.

49. (Previously Presented) The method of claim 44 further comprising:

receiving a second program read request from a second CPU at a second line cache interface;

generating a second address from said second program read request; and

resolving line cache access conflicts between said first and second CPUs.

50. (Previously Presented) The method of claim 44 further comprising:

storing data in line cache memory;

storing addresses associated with said data stored in said line cache memory;

determining when one of a hit and a miss occurs; and

managing retrieval of data from said first and second memory devices when said miss occurs.

51. (Currently Amended) A method for operating a line cache, comprising:

receiving a first program read request from a first CPU at a first line cache interface;

generating a first address from said first program read request;

receiving a second program read request from a second CPU at a second line cache interface;

generating a second address from said second program read request;

resolving line cache access conflicts between said first and second CPUs;

selectively connecting said line cache to one of first and second memory interfaces for first and second memory devices, respectively;

comparing said first address to stored addresses in said line cache when said line cache receives said first address;

returning data associated with said first address if a match occurs;

retrieving data from one of said first and second memory devices if a miss occurs;

storing data in line cache memory;

storing addresses associated with said data stored in said line cache memory;

determining when one of a hit and a miss occurs; and

managing retrieval of data from said first and second memory devices when said miss occurs,

wherein said line cache memory includes multiple pages and further comprising allowing one page to be accessed by one of said first CPU and said second CPU while the other of said first CPU and said second CPU is waiting for data retrieval in another page, and

wherein said selectively connecting includes selectively connecting said line cache to one of said first and second memory interfaces with a plurality of selectors that each receive a second address based on the first address and each select between first and second sets of signals relating to the first and second memory devices, respectively, based on the second address.

52. (Previously Ppresented) The method of claim 50 further comprising identifying a first least used page in said line cache.

53. (Previously Presented) The method of claim 52 further comprising replacing said first least used page with data retrieved from one of the first and second memory devices when a miss occurs.

54. (Previously Ppresented) The method of claim 50 further comprising operating said line cache based on at least one internal state of said first CPU.

55. (Currently Amended) A method for operating a line cache, comprising:

receiving a first program read request from a first CPU at a first line cache interface;

generating a first address from said first program read request;

selectively connecting said line cache to one of first and second memory interfaces for first and second memory devices, respectively, wherein said selectively connecting includes selectively connecting said line cache to said one of said first and second memory interfaces with a plurality of selectors that each receive a second address based on the first address and each select between first and second sets of signals relating to the first and second memory devices, respectively, based on the second address;

comparing said first address to stored addresses in said line cache when said line cache receives said first address;

returning data associated with said first address if a match occurs;

retrieving data from one of said first and second memory devices if a miss occurs;

storing data in line cache memory;

storing addresses associated with said data stored in said line cache memory;

determining when one of a hit and a miss occurs;

managing retrieval of data from said first and second memory devices when said miss occurs;

identifying a first least used page in said line cache;

identifying a second least used page; and

checking at least one internal state of said first CPU.

56. (Previously Presented) The method of claim 55 further comprising replacing said first least used page when a miss occurs and said at least one internal state of said first CPU do not indicate a likelihood that said first least used page will be needed within a predetermined period.

57. (Previously Presented) The method of claim 55 further comprising replacing said second least used page when a miss occurs and said at least one internal state of said first CPU indicate a likelihood that said first least used page will be needed within a predetermined period.

58. (Currently Amended) A method for operating a line cache, comprising:

receiving a first program read request from a first CPU at a first line cache interface;

generating a first address from said first program read request;

selectively connecting said line cache to one of first and second memory interfaces for first and second memory devices, respectively, wherein said selectively connecting includes selectively connecting said line cache to said one of said first and second memory interfaces with a plurality of selectors that each receive a second address based on the first address and each select between first and second sets of signals relating to the first and second memory devices, respectively, based on the second address;

comparing said first address to stored addresses in said line cache when said line cache receives said first address;

returning data associated with said first address if a match occurs; and

retrieving data from one of said first and second memory devices if a miss occurs,

wherein said first CPU executes an application and further comprising basing a line width and number of pages of said line cache on said application.

59. (Currently Amended) A method for operating a line cache, comprising:

receiving a first program read request from a first CPU at a first line cache interface;

generating a first address from said first program read request;

receiving a second program read request from a second CPU at a second line cache interface;

generating a second address from said second program read request;

generating a translated address based on at least one of the first address and the second address, wherein the translated address includes a memory select portion;

resolving line cache access conflicts between said first CPU and said second CPU; and

selectively connecting said line cache to one of first and second memory interfaces for first and second memory devices, respectively, based on the memory select portion, wherein said selectively connecting includes selectively connecting said line cache to one of said first and second memory interfaces with a plurality of selectors that each receive the memory select portion and each select between first and second sets of signals relating to the first and second memory devices, respectively, based on the memory select portion.

60. (Previously Presented) The method of Claim 59 further comprising:

selecting one of said first and second addresses;

comparing said selected one of said first and second addresses to stored addresses in said line cache;

returning data associated with said selected one of said first and second addresses if a match occurs; and

retrieving data from one of said first and second memory devices if a miss occurs.

61. (Previously Presented) The method of Claim 59 wherein said first memory device is random access memory (RAM) and said second memory device is flash memory.

62. (Previously Presented) The method of Claim 59 wherein said first CPU is an embedded processor.

63. (Previously Presented) The method of Claim 59 further comprising:

storing data in line cache memory;

storing addresses associated with said data stored in said line cache memory;

determining when one of a hit and a miss occurs; and

managing retrieval of data from said first and second memory devices when said miss occurs.

64-78. (Cancelled)

79. (Currently Amended) A line cache control system comprising:

first processing means for processing data;

first and second memory means for storing data;

first line cache interface means that is associated with said first processing means for receiving a first program read request from said first processing means and for generating a first address from said first program read request;

first memory interface means for communicating with said first memory means;

second memory interface means for communicating with said second memory means;

line cache means for storing data and receiving a translated address based on the first address that includes a memory select portion; and

selecting means for receiving the translated address and selectively connecting said line cache means to one of said first and second memory interface means based on the memory select portion,

wherein when said line cache means receives said translated address, said line cache means compares said translated address to stored addresses in said line cache means, returns data associated with said translated address if a match occurs, and retrieves data from one of the first and second memory means if a miss occurs, and

wherein said selecting means includes a plurality of selectors that each receive the translated address and each select between first and second sets of signals relating to the first and second memory means, respectively, based on the translated address.

80. (Previously Presented) The line cache control system of claim 79 wherein

said first memory means is random access memory (RAM).

81. (Previously Presented) The line cache control system of claim 80 wherein said RAM is one of dynamic RAM (DRAM), synchronous DRAM (SDRAM), and double data rate SDRAM (DDRAM).

82. (Previously Presented) The line cache control system of claim 79 wherein said second memory means is flash memory.

83. (Previously Presented) The line cache control system of claim 79 wherein said first processing means is an embedded processor.

84. (Previously Presented) The line cache control system of claim 79 further comprising:

second processing means for processing data;

second line cache interface means that is associated with said second processing means, that receives a second program read request from said second processing means and that generates a second address from said second program read request; and

line cache arbitration means that communicates with said first and second line cache interface means for resolving access conflicts to said line cache means between said first processing means and said second processing means.

85. (Previously Presented) The line cache control system of claim 84 further comprising:

first direct interface means that is associated with said first processing means,

wherein said first memory interface means includes a second direct interface means that communicates with said first direct interface means and wherein said first and second direct interface means allow said first processing means to at least one of read and write data directly to said first memory means.

86. (Previously Presented) The line cache control system of claim 85 further comprising:

third direct interface means that is associated with said second processing means,

wherein said second memory interface means includes a fourth direct interface means that communicates with said third direct interface means and wherein said third and fourth direct interface means allow said second processing means to at least one of read and write data directly to said second memory means.

87. (Previously Presented) The line cache control system of claim 86 further comprising direct read/write arbitration means for resolving direct memory access conflicts between said first and third direct interface means.

88. (Previously Presented) The line cache control system of claim 84 wherein

said first processing means is a host processor for a hard disk drive and said second processing means is a servo processor for said hard disk drive.

89. (Previously Presented) The line cache control system of claim 79 wherein said line cache means includes:

data storing means for storing data;

content addressable memory means for storing addresses associated with said data stored in said data storing means; and

line cache control means that determines when one of a hit and a miss occurs and that manages retrieval of data from said first and second memory means when said miss occurs.

90. (Currently Amended) A line cache control system comprising:

first processing means for processing data;

first and second memory means for storing data;

first line cache interface means that is associated with said first processing means for receiving a first program read request from said first processing means and for generating a first address from said first program read request;

first memory interface means for communicating with said first memory means;

second memory interface means for communicating with said second memory means;

line cache means for storing data that includes:

data storing means for storing data;

content addressable memory means for storing addresses associated with said data stored in said data storing means; and

line cache control means that determines when one of a hit and a miss occurs and that manages retrieval of data from said first and second memory means when said miss occurs; and

selecting means for selectively connecting said line cache means to one of said first and second memory interface means,

wherein when said line cache means receives said first address, said line cache means compares said first address to stored addresses in said line cache means, returns data associated with said first address if a match occurs, and retrieves data from one of the first and second memory means if a miss occurs; ~~and~~

wherein said line cache means includes multiple pages and wherein said line cache control means allows one page to be accessed by one of said first and second processing means while the other of said first and second processing means is waiting for data retrieval in another page; and

wherein said selecting means includes a plurality of selectors that each receive a second address based on the first address and each select between first and second sets of signals relating to the first and second memory means, respectively, based on the second address.

91. (Previously Presented) The line cache control system of claim 89 further comprising least used page means for identifying a first least used page in said line

cache means.

92. (Previously Presented) The line cache control system of claim 91 wherein said line cache control means replaces said first least used page with data retrieved from one of the first and second memory means when a miss occurs.

93. (Previously Presented) The line cache control system of claim 92 wherein said least used page means identifies a second least used page and wherein said line cache control means checks internal states of said first processing means.

94. (Currently Amended) A line cache control system comprising:

first processing means for processing data;

first and second memory means for storing data;

first line cache interface means that is associated with said first processing means for receiving a first program read request from said first processing means and for generating a first address from said first program read request;

first memory interface means for communicating with said first memory means;

second memory interface means for communicating with said second memory means;

line cache means for storing data that includes:

data storing means for storing data;

content addressable memory means for storing addresses

associated with said data stored in said data storing means; and

line cache control means that determines when one of a hit and a miss occurs and that manages retrieval of data from said first and second memory means when said miss occurs;

selecting means for selectively connecting said line cache means to one of said first and second memory interface means;

least used page means for identifying a first least used page in said line cache means

wherein when said line cache means receives said first address, said line cache means compares said first address to stored addresses in said line cache means, returns data associated with said first address if a match occurs, and retrieves data from one of the first and second memory means if a miss occurs, said line cache control means replaces said first least used page with data retrieved from one of the first and second memory means when a miss occurs, said least used page means identifies a second least used page and wherein said line cache control means checks internal states of said first processing means, and said first least used page is replaced when a miss occurs and internal states of said first processing means do not indicate a likelihood that said first least used page will be needed within a predetermined period, and

wherein said selecting means includes a plurality of selectors that each receive a second address based on the first address and each select between first and second sets of signals relating to the first and second memory means, respectively, based on the second address.

95. (Previously Presented) The line cache control system of claim 94 wherein said second least used page is replaced when a miss occurs and internal states of said first processing means indicate a likelihood that said first least used page will be needed within a predetermined period.

96. (Currently Amended) A line cache control system comprising:

first processing means for processing data;

first and second memory means for storing data;

first line cache interface means that is associated with said first processing means for receiving a first program read request from said first processing means and for generating a first address from said first program read request;

first memory interface means for communicating with said first memory means;

second memory interface means for communicating with said second memory means;

line cache means for storing data; and

selecting means for selectively connecting said line cache means to one of said first and second memory interface means,

wherein when said line cache means receives said first address, said line cache means compares said first address to stored addresses in said line cache means, returns data associated with said first address if a match occurs, and retrieves data from one of the first and second memory means if a miss occurs, and said first processing means executes an application and wherein said line cache has a line width

and number of pages that are based on said application, and

wherein said selecting means includes a plurality of selectors that each receive a second address based on the first address and each select between first and second sets of signals relating to the first and second memory means, respectively, based on the second address.

97. (Previously Presented) The line cache control system of claim 79 wherein said line cache includes 4 pages of 8 x 32 bits.

98. (Currently Amended) A memory control system for a line cache and first and second memory devices that are accessed by a first central processing unit (CPU) and a second CPU, comprising:

first line cache interface means that is associated with the first CPU, for receiving a first program read request from the first CPU and for generating a first address from said first program read request;

second line cache interface means that is associated with the second CPU, for receiving a second program read request from the second CPU and for generating a second address from said second program read request;

line cache arbitration means that communicates with said first and second line cache interface means and said line cache for resolving line cache access conflicts between the first CPU and the second CPU and for generating a translated address based on one of the first address and the second address that includes a memory select portion; and

switching means for receiving the translated address and for selectively connecting said line cache to one of said first and second memory devices based on the memory select portion, wherein said selecting means includes a plurality of selectors that each receive the translated address and each select between first and second sets of signals relating to the first and second memory device, respectively, based on the translated address.

99. (Previously Presented) The memory control system of Claim 98 wherein said line cache arbitration means selects one of said first and second addresses and further comprising:

first memory interface means for communicating with the first memory device; and

second memory interface means for communicating with the second memory device, wherein when said line cache receives said selected one of said first and second addresses from said line cache arbitration means, said line cache compares said selected one of said first and second addresses to stored addresses in said line cache, returns data associated with said selected one of said first and second addresses if a match occurs, and retrieves data from one of the first and second memory devices if a miss occurs.

100. (Previously Presented) The memory control system of claim 98 wherein the first memory device is random access memory (RAM) and the second memory device is flash memory.

101. (Previously Presented) The memory control system of claim 98 wherein the first CPU is an embedded processor.

102. (Previously Presented) The memory control system of claim 98 further comprising:

first direct interface means that is associated with the first CPU, wherein said first memory interface means includes a second direct interface means for communicating with said first direct interface means and wherein said first and second direct interface means allow the first CPU to at least one of read and write data directly to the first memory device; and

a third direct interface means that is associated with the second CPU, wherein said second memory interface means includes a fourth direct interface means for communicating with said third direct interface means and wherein said third and fourth direct interface means allow the second CPU to at least one of read and write data directly to the second memory device.

103. (Previously Presented) The memory control system of claim 102 further comprising direct read/write arbitration means for resolving direct memory access conflicts between said first and third direct interface means.

104. (Previously Presented) The memory control system of claim 98 wherein the first CPU is a host processor for a hard disk drive and the second CPU is a servo processor for said hard disk drive.

105. (Previously Presented) The memory control system of claim 98 wherein said line cache includes:

data storing means for storing data;

content addressable memory (CAM) means for storing addresses associated with said data stored in said storing means; and

line cache control means for determining when one of a hit and a miss occurs and for managing retrieval of data from the first and second memory devices when said miss occurs.

106-120 (Cancelled).